(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) Publication number: **0 121 291**
A2

(12) # EUROPEAN PATENT APPLICATION

(21) Application number: 84300228.8

(22) Date of filing: 16.01.84

(51) Int. Cl.³: H 04 M 3/50
H 04 M 11/10, G 06 F 3/06

(30) Priority: 30.03.83 US 480906

(43) Date of publication of application:
10.10.84 Bulletin 84/41

(84) Designated Contracting States:
AT BE CH DE FR GB IT LI LU NL SE

(71) Applicant: VOICETEK CORPORATION
10 Dedham Street
Newton Massachusetts 02161(US)

(72) Inventor: Szeto, Charles
38 Ellis Street
Medway Massachusetts 02053(US)

(74) Representative: Allsop, John Rowland et al,
Combes Gate 96 Ellesborough Road
Wendover, Bucks., HP22 6EW(GB)

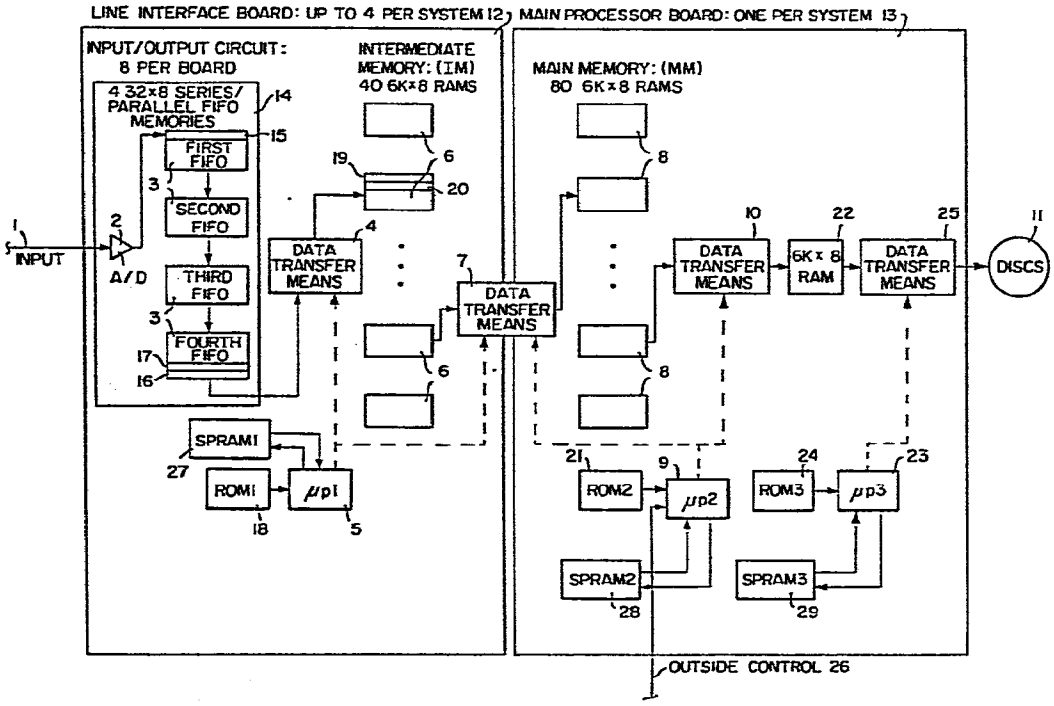
(54) **Method of and apparatus for voice communication storage and forwarding with sumultaneous access to multiple users.**

(57) This technique for voice or other similar communication signal storage (such as recording) and forwarding (such as playback) enables simultaneous access to multiple users through use of first-in-first-out memories (FIFO), microprocessor-controlled to fill intermediate memory RAMs at irregular intervals and to transfer the contents to main memory RAMs for ultimate disc recording, with playback therefrom microprocessor-controlled at irregular intervals to transfer back recorded data blocks through the main memory RAMs and appropriate intermediate memory RAMs and FIFO memories associated with the appropriate playback channel or user; the various data transfers being controlled to avoid delays in the played back messages that are long enough to be perceived by the channel user, and with succeeding segments of data being transferred to intermediate memory RAMs by the time the transfer of the previous segment is complete, thereby to insure continuity of communication.

EP 0 121 291 A2

./...

FIG. 1.


LINE INTERFACE BOARD: UP TO 4 PER SYSTEM 12
MAIN PROCESSOR BOARD: ONE PER SYSTEM 13
INPUT/OUTPUT CIRCUIT: 8 PER BOARD
4 32x8 SERIES/ PARALLEL FIFO MEMORIES
FIRST FIFO
SECOND FIFO
THIRD FIFO
FOURTH FIFO
INPUT
A/D
INTERMEDIATE MEMORY: (IM) 40 6K x 8 RAMS
DATA TRANSFER MEANS
SPRAM1
ROM1
μp1
MAIN MEMORY: (MM) 80 6K x 8 RAMS
ROM2
μp2
SPRAM2
6K x 8 RAM
ROM3
μp3
SPRAM3
DISCS
OUTSIDE CONTROL 26

METHOD OF AND APPARATUS FOR VOICE COMMUNICATION STORAGE AND FORWARDING WITH SIMULTANEOUS ACCESS TO MULTIPLE USERS

The present invention relates to voice communication storage and retrieval systems as of the type in which a centralized repository is employed to receive and store voice communication messages from multiple users and to be accessed for the retrieval or forwarding of the messages by telephone; the invention being more particularly concerned with the preferred employment of radio-actuated paging; and with permitting simultaneous access to a number of users placing and retrieving messages without subjecting them to delays.

It has been proposed to employ a plurality of magnetic tape recorders for receiving, serially recording and enabling accessing of messages, with complex and expensive multiplexing for achieving simultaneous communication for the above purposes, as described, for example, in U.S. Patent No. 4,260,854. Such systems are, however, seriously limiting in application by virtue of the fact that recorded messages on tape cannot be accessed randomly and are really useful only when one associates one tape and one recording machine with one individual or one tele-

phone. Further, the recording medium cannot be accessed simultaneously by more than one user; such simultaneous access being a particularly important attribute for a centralized system serving many users at a busy time of the day, and a feature attained by the present invention.

While other approaches have involved disc-based systems with complex storage techniques such as pairs of basic memory buffers for recording on one side while transferring data from the other; as in U.S. Patent No. 4,371,752, such are limited either by inability to record conversations of random length (from seconds to hours) and/or by expensive construction--limitations overcome by the present invention through its economy of construction residing in the use of dynamically assigned blocks of memory under novel microprocessor control.

In accordance with the invention, it is now possible to create such a system at reasonable cost readily adaptable so that messages may be placed and retrieved by telephone for applications such as the following:

1. Centralized telephone answering capability for a large number of users, typically employees of an organization. The equipment is readily adaptable to allow messages to be forwarded to other system users than the user for whom the message was originally intended; such message forwarding being passive or forced. Thus messages may be left by anyone for subscribers to the system and may be retrieved by subscribers to the system at any convenient time. Such operation is sometimes described as "voice mail".
2. Telephone answering services can be automated using this approach.
3. Radio-actuated paging system can provide automatic recording of voice messages which can than be accessed by the individual pages (tone only) when such individual can reach a telephone.
4. Some specialized dictation systems such as those used to dictate medical test reports or the like can use the system; the information being accessed by listening to the voice recording as soon as the report is dictated, without waiting for the transcription.

An object of the invention, accordingly, is to provide a new and improved method of and apparatus for voice communication storage and forwarding that is not subject to the prior art limitations, above-mentioned and others, but that enables simultaneous access to multiple users with great flexibility of application and low cost.

A futher object is to provide a novel message storing and retrieval technique of more broad utility, as well.

Other and further objects will be explained hereinafter and are more particularly delineated in the appended claims.

The method underlying the invention converts the electrical signal which is an analog of the sound pressure of the voice to a digital signal and stores the digital data on a magnetic disc (a Winchester or hard disc), accessable at random such that any communication can be rapidly located and played back. Further, as explained more fully hereinafter, because of the speed with which the data can be transferred to the disc and retrieved from the disc, a buffering system is provided which allows a number of users (up to 32 in the preferred embodiment) to use the same

disc at the same time, each with the impression that the disc is totally at the user's disposal (i.e., simultaneous access).

In summary, from one of its important aspects, the invention embraces a method of message communication digitized storage (in recording mode) and forwarding (in playback mode) along a plurality of input/output channels with simultaneous access to multiple users with the aid of a plurality of line interface systems having series/parallel first-in-first-out memories (FIFO) and intermediate memory RAMS under first microprocessor control and with data transfer connection to main memory RAMS the contents of which are data transferable under second microprocessor control with memory discs, said method comprising, reading the message recording or playback status of each of the plurality of channels in order; if in a recording mode, determining the state of filling of the FIFO memories and, under the control of the first microprocessor, transferring the data in as many filled FIFO memories as exist at such time to the first available empty data block of the intermediate memory RAMS, and

when such block is filled, transferring the data thereof to the main memory RAMS; if in a playback mode, determining the state of filling of the successive FIFO memories and transferring channel data blocks to fill empty FIFO memories with message data being played back; examining the recording or playback status of each input/output channel of the plurality of line interface systems under control of the second microprocessor; receiving commands identifying which channels are assigned for recording and which for playback; if a recording mode is assigned, and the said examining of said status has recognized such recording mode, determining whether data blocks from such channel still exist in the main memory and transferring such to the memory discs; if the recording mode is not recognized during such examining, communicating such status to the first microprocessor of the appropriate channel line interface system; if a playback mode is assigned, and the said examining of such status has recognized such playback mode, determining whether the channel is calling for more data, and if so, transferring data blocks of such channel from full

main memory RAMS to intermediate memory RAMS and successive FIFO memories of the appropriate channel line interface system for playback; and, if the playback mode is not recognized during such examining, communicating an interrupt to the first microprocessor of the appropriate line interface system. Preferred and best mode embodiments and details of construction and operation are hereinafter presented.

The invention will now be described in connection with the accompanying drawings. Fig. 1 of which is a block circuit diagram of a preferred embodiment of the invention illustrating data flow and control of data transfer for a voice or similar storage and forwarding system with simultaneous access for multiple users, shown operating in the recording mode;

Fig. 2 is a similar diagram of the system in playback operation mode;

Fig . 3 is a more detailed diagram of the line interface input/output circuits employable in the systems of Figs. 1 and 2;

Fig. 4 is a similar detailed diagram of the processor and storage section of each of the line interface systems or boards;

Fig. 5 illustrates circuit details for transferring of data between the line interface systems or circuit boards and the main processor board of Figs. 1 and 2;

Fig. 6 is a diagram of the details of such main processor boards with address, control, and some circuitry omitted for clarity and to highlight the novel features of the invention; and

Figs. 7 and 8 are simplified flow charts illustrating the programming operational steps employed by the microprocessors of each line interface board and main processor board, respectively.

Referring to Fig. 1, an overview of a preferred embodiment is illustrated operating to record speech. Data flow paths are shown with solid arrowed lines and data transfer, with dashed lines. The system is provided with hard discs 11, a main processor board 13, and from one to four line interface boards 12. Each of the line interface boards contains eight independent inputs (or outputs) for audio signals 14, one of which is shown in Fig. 1.

In the recording mode, each input 1 is supplied with a voltage signal which is an analog of the voice

sound pressure applied to the microphone of a telephone transmitter, for example. This signal is converted to a digital signal by an analog-to-digital converter (A/D) 2. The preferred A/D conversion scheme is the continously variable slope delta (CVSD) modulation method in which the continuously varying audio signal is converted into a bit stream. At the preferred sampling rate of say 20 kHz, the output of the A/D converter consists of a series of voltage pulses, spaced apart by 50 microseconds. Each pulse has a height of 5 V (a "one") or of 0 V (a "zero"), depending on whether the audio signal slope increased or decreased in amplitude within the previous 50 microseconds. This bit stream can be reconstituted into an analog voltage representation of the sound when applied to the input of an appropriate digital to analog (D/A) converter, later discussed.

The storage problem thus resolves to the problem of storing the bit stream produced by the continously variable slope data modulation A/D converter. The first step in the storage process is the assembly of the bit stream into a series of, for example, 8-bit words. Each of the A/D converters of the line inter-

face boards 12 is shown connected to a series of four 32x8 bit serial/parallel, first-in-first-out (FIFO) memories 3. As the bit stream begins to flow it fills the top register 15 of the first 32x8 FIFO memory. When the top register of the first FIFO is filled, the contents are shifted ("rippled") immediately to the bottom register of the fourth FIFO memory 16. The top register of the first FIFO memory is then free to assemble the next 8-bit word from the data stream. When the register is filled again, the resultant word is transferred to the second-from-bottom register in the fourth FIFO memory 27. In the same way, the top register of the first FIFO memory keeps assembling the bit stream into 8 bit words which are passed down the FIFO memories to the last available unfilled register. At irregular intervals, under control of the program stored in read-only-memory ROM1 18 and scratchpad SPRAM1 27, microprocessor (µp1) 5 investigates the state of affairs of the FIFO memories 3. When at least the fourth of these FIFO memories is filled, µp1 causes the contents to be transferred to one of the forty 6kx8 (6,144x8) random access memories (RAM) 6 (the intermediate memory) using conventional data transfer means 4.

The choice of which of the RAMs 6 of intermediate memory (IM) is to be used is determined by μpl according to the program hereinafter described. At the time a decision to transfer is made, the contents of all filled FIFO memories will be transferred; that is, the contents of the third (if it is full) and the contents of the second (if it is full), in addition to the contents of the fourth. The number of FIFO memories and the interval at which μpl makes transfers does not allow the first FIFO memory to ever fill up, since if all the FIFO memories were to fill before a data transfer, data would be lost.

Once data from the filled FIFOs is transferred, the data in the partially filled FIFO remaining above the last FIFO to be emptied is rippled to the bottom of the fourth FIFO, and the process of accumulating data in the FIFO registers continues.

The transfer of data to one of the RAMs 6 is organized so that at the first data transfer, the contents of the bottom register 16 in the fourth FIFO memory is placed in the top register 19 of the RAM; the contents of the second-to-bottom register 17 of the fourth FIFO memory is placed in the second register 20 of the RAM/ and so forth. Thus the oldest

data is at the bottom of the series of FIFO memories; and the oldest data is at the top of the RAM.

Once an assignment is made of the RAM 6 to be used for recording data from a particular audio input 1, µpl keeps adding data from that source until the RAM is filled. Thus the RAM accumulates 6kx8 bits of data. Since the bits are produced one each 50 microseconds, each RAM will contain $6144 \times 8 \times 50 \times 10^{-6} = 2.5$ seconds of conversation. Subsequent 2.5 seconds of conversation are stored in other RAMs 6 as assigned by µpl 5.

The main processor board 13 provides an assembly point for data from the (up to) four line interface boards 12. As the intermediate memory RAMs 6 on the line interface boards fill up, the µpl 5 on each board queues the filled RAMs for transfer to the eighty 6kx8 RAMs 8 on the main processor board--the main memory M M. At irregular intervals, under control of the program stored in ROM2 21 and scratchpad SPRAM2 28, the further microprocessor µp2 9, shown on the main processor board 13, investigates the state of affairs of the queues as reported by the µpl 5 of each of the line interface boards 12, and causes the contents of

the RAM 6 at the head of the queue to be transferred to one of the RAMs 8 using conventional data transfer means 7. The choice of which of the RAMs 8 is to be used is determined by μp2 according to the later-described program. Once the contents of a RAM 6 are transferred, the RAM is returned to the pool of unused RAMs and may be reassigned by μp1 5.

As the RAMs 8 in the main processor board 13 fill up with data transferred from the audio interface boards 12, the microprocessor μp2 queues the filled RAMs 8 for data transfer to the discs 11. The contents of the RAM 8 at the head of the queue are transferred at irregular intervals by μp2 using data transfer means 10 to a single 6kx8 RAM 22. The data is then transferred using data transfer means 25 from RAM 22 to a disc under the control of μp3 23. This microprocessor receives its instructions from ROM3 24 which contains the disc control program and from scratchpad SPRAM3 29. Once the contents of one of the main memory RAMs 8 are transferred to the disc system, the RAM is returned to the pool of unused RAMs and may be reassigned by μp2. System response is initiated by outside control 26 to μp2, illustrated at the bottom of Fig. 1.

The operation of the system to play back speech is shown in Fig. 2, with components similar to those of Fig. 1 identified by prime notations.

In the playback mode, the system is instructed (by external means 26) to play back a message previously recorded. As indicated above, the message has been stored in 2.5 second segments on the discs 11'. The disc control system, under control of up3 23' and the program stored in ROM3 6' and SPRAM 3 29', finds the segment of disc containing the first part of the 2.5 seconds of the message to be played back, and transfers the data to a 6kx8 RAM 22' using data transfer means 25'. At irregular intervals, up2 9' under control of ROM2 21' and SPRAM 28' assigns one of the available 6kx8 RAMs 8' in the main memory to receive this data. The choice of RAM is made by the program later described. Data is then transferred to the RAM 8' from RAM 22' using the data transfer.

At irregular intervals, up2 9' recognizes the existence of data in one of the RAMs 8' waiting transfer out. The up2 9' recognizes which of the (up to) four line interface boards 12' contains the audio channel to which the playback is directed and trans-

fers the data from the RAM 8' to one of the available RAMs 6' on the appropriate line interface board, using data transfer means 7'.

At irregular intervals, μpl 5', under control of the program stored in ROM1 18' and SPRAM1 27', recognize the presence of the first (and subsequent) 2.5 second segments of speech in one of the RAMs 6' and which of the 8 input/output circuits 14' is to be used. Data has been stored in the RAMs such that the first (oldest) data to be played back is in the top register 37 of the RAM; using data transfer means 4', the 8-bit word in the top register 37 of the RAM is transferred to the top register 15' of the first of the four 8x32 FIFO memories 3'. As in the case of the record mode (Fig. 1), this word is automatically transferred to the bottom register 16' of the fourth FIFO memory. The next 8-bit word at register 38 in the RAM 6' is then transferred to the top register of the first FIFO memory; this word is then rippled down through the four FIFO memory. As this process continues, the contents of the bottom register of the fourth FIFO memory is removed bit-by-bit (oldest bit first) at the 20 kHz rate and fed to the input of a

D/A converter 2'. As soon as the contents of the bottom register 16' have been transferred, the contents of the next higher register 17' are transferred to the bottom register 16', allowing the D/A conversion to continue at the same 20 kHz rate used to digitize the data.

System timing is so arranged that the series of four FIFO memories is never completely empty, thus avoiding delays in the played back conversations that are long enough to be perceived by the listener. Further, at least the next segment of 2.5 second conversation has been transferred to one of the RAMs 6' by the time the transfer of the previous segment is complete, ensuring continuity.

The D/A converter 2' reconstitutes the analog audio signal using the 20 kHz bit stream, employing the bits to control the slope of a continuously varying audio output signal; the process being opposite to that used to digitize the signal by the CVSD modulation method described in Fig. 1. The audio output appears on the output line so-labelled to the left in Fig. 2.

As with the operation in the record mode (Fig. 1), once the contents of RAM 6' or 8' have been transferred, the RAM is reassigned to a pool of available memory.

The process of recording described in connection with Fig. 1 and of playback described in connection with Fig. 2 may proceed at the same time with up to, for example, 32 channels operating simultaneously with any number n of channels devoted to recording and the balance of the channels (32-n) devoted to playback. The number of channels devoted to each service and the number idle at any moment is determiend by the external calls for service at the moment. Once a channel which has been recording or playing back is released, it enters a pool of idle channels which can be reassigned for either recording or playback at the next call for service.

Channels may, if desired, be alternated between playback and record during the course of a conversation, allowing for the insertion of prerecorded prompts which help the user to operate the system. In a typical application, a 32-channel system permits reasonable service for several thousand subscribers.

It is now in order to examine in more detail the operation and programs used to control the system.

The details of the input/output circuit, eight of which are included on each line interface board, are shown in Fig. 3. Clock, some circuitry, and control signals are not shown for clarity and in order not to detract from the essential features of novelty underlying the invention. The most important control signal (provided by µpl) defines whether the input/output circuit will operate in the record or in the playback mode.

When directed to operate in the record mode (Fig. 1), the audio line 1 is routed through a solid state analog switch S to the appropriate input of an antialiasing filter F. In this mode of operation, the filter serves as a bandpass filter with a pass band of from, say, 100 Hz to 3500 Hz, thereby reducing any extraneous noise components on the input signal. The signal is then passed to the A/D CVSD modulator 2 which provides a bit stream at the 20 kHz clock rate. A bus connector is shown at BC1 conected to the FIFO memories 3, being set to a high impedance state; while a further bus connector BC2, connected to the A/D

modulator 2 and to the bus connector BC1, is set to pass the bit stream to the first series/parallel FIFO memory 3. The bit stream is formed into 8-bit words as described above, and the FIFO memories 3 are connected so as to ripple each word as it is formed to the bottom-most available register of the series of FIFO memories 3. When one of the FIFO memories is filled, a flag is set which may be passed to μpl 5 through bus connector BC3, upon command. When μpl 5 decides to transfer data, the lower bus connector BC4 connected to the lowermost or fourth FIFO 3, is enabled, placing the contents of the bottom register of FIFO memory onto the data bus DB. The μpl 5 continues to extract data, which keeps rippling to the bottom register until all the contents of the filled FIFO memories 3 are transferred.

When directed to operate in the playback mode, bus connector BC2 is set to the high impedance state and μpl 5 begins to place a series of 8-bit words on the data bus DB. As each word is placed in the top register of the FIFO memory through bus connector BC1, it ripples to the bottom register of FIFO memory 3 and is extracted, bit by bit, at the 20 kHz clock rate,

bus connectors BC3 and BC4 being in the high impedance state. The bit stream is passed to the D/A converter of the modulator 2', the resultant audio signal is smoothed by the filter F, and is passed on to the audio line 1' by the switch S which has now been set to the "out" or playback position.

Fig. 4 illustrates details of the microprocessor µpl 5, its associated read-only memory ROM1 18, a direct memory access controller DMA1, the scratchpad SPRAM1 27, and the intermediate random access memory 6 used for storage of 2.5 second segments of speech by each of the (up to) four line interface boards 12 in each system. Address, control, and some connecting circuitry are again omitted for clarity.

When the µpl 5 decides to transfer data from one of the eight channels on its board which is in operation in the record mode, Fig. 1, it first determines how many of the FIFO memories 3 used by the channel are filled. If only the bottom FIFO is filled (FIFO 4 in Fig. 3), µpl 5 transfers the contents to intermediate memory 6 by making 32 transfers through an internal register. If more than one of the FIFOs is filled, µpl 5 transfers the data using a faster

method. It first sets up DMA1 to effect a direct memory access transfer, and then turns over the transfer to DMA1 for 64 or 96 transfers, corresponding to a filled condition of FIFOs numbers 3 and 4 or FIFOs 2, 3, and 4 (see Fig. 3). Since it takes longer to set up DMA1 than to transfer the data directly from one FIFO, this approach results in optimum speed of transfer.

The intermediate memory 6 is shown organized as an array of thirty-two 64kx1 RAMs. Each of the four rows of RAMs contains eight columns of RAMs, one for each bit in the eight bit words being transferred from the FIFOs. The memory is organized into 40 segments, each of 6k 8-bit words (2.5 seconds of conversation). The µpl 5 identifies an available segment of the intermediate memory in which to place each 2.5 second segment of conversation from the channel providing the input. When a 2.5 second segment of intermediate memory is filled, a new segment is identified and filled with the next 2.5 seconds of conversation. Bus connector BC5 is enabled to take data from µpl's data bus DB and place them into the appropriate RAMs as directed by the address signals from µpl 5 or from the

DMA1 controller, whichever is responsible for the transfer.

When the channel being serviced is designated as a playback channel (Fig. 2), the μpl's task is to keep the FIFOs associated with this channel supplied with data from the message which is to be played back. Subsequent circuitry will keep at least one 6k segment of intermediate memory available for transfer to the FIFOs associated with the channel being serviced. If only one of the FIFOs (FIFO 1 of Fig. 3) is empty, transfer will be made through μpl 5. If more than one FIFO of Fig. 3 is empty, transfer will be made by DMA1, thereby optimizing speed of transfer as in the case of data recording. Bus connector BC5 is enabled to take data from μpl's data bus DB as supplied by the appropriately addressed RAMS and supply it to the FIFOs.

The system is fast enough to allow all eight input/output circuits to be served. That is, if all channels are busy, no combination of input/output channels exist such that the FIFOs of any recording channel never fill up before data transfer occurs to clear them out to intermediate memory IM, and the

FIFOs of no playback channel are never totally emptied before data is supplied to replenish them. This condition is fulfilled by the architecture described, together with the use of a microprocessor µpl capable of rapid operation and programming which emphasizes processing speed (for instance using alternative direct or DMA transfer).

Details of how voice data may be transferred from each of the (up to) four line interface boards 12 to the main processor board 13 are illustrated in Fig. 5. The data transfer requires special means because the line interface boards are operating under control of their microprocesors µpl 5 and the main processor board is operating under control of its completely independent microprocessor µp2 9, Fig. 1. Each of the line interface boards is equipped wth two 6kx8 RAMs 8" and 8'". The first of the RAMs 8" is used to transfer data in 6k blocks (2.5 seconds of conversation) from one of the blocks of data in intermediate memory IM (Fig. 4) to the main memory MM on the main processor board (record mode, Fig. 1). The second of the RAMs 8'" is used to transfer data in 6k blocks from one of the blocks in the main memory MM to

one of the available blocks in the intermediate memory IM of the apropriate line interface board (playback mode, Fig. 2).

Data transfer out to the main processor board 13 (record mode, Fig. 1) is initiated by µpl 5 which queues 6k blocks of data in IM for transfer. The µpl 5 takes the block at the head of the queue and using controller DMA1, Fig. 4, transfers the data to the RAM by placing it on its data bus DB. Bus connector BC7 is opened and bus connector BC9 is closed by µpl 5 to allow the shared data bus SB' to act as an extension of the data bus DB of µpl 5. At the same time, µpl 5 provides a signal to µp2 9 indicating that a block of data is ready for transfer (service request). When µp2 9 is ready to service the request, it makes shared bus SB' an extension of its data bus DB' by turning bus connector BC9 on, and bus connector BC7 off. The µp2 9 then effects a transfer using its DMA2 controller, now to be discussed in connection with Fig. 6.

Data transfer from the main processor board (playback mode, Fig. 2) is initiated by µp2 9 on the main processor board. The transfer of 6k blocks to RAM 8' is performed by µp2 9 using its DMA2 control-

ler, Fig. 6, and gaining access to the shared bus SB (Fig. 5) by turning bus connector BC8 on, and bus connector BC6 off. Once the transfer is complete, µp2 9 provides a signal to the appropriate µp1 5 and this upl transfers the contents of RAM 8' to an available block of 6k IM in its board, using its DMA1 controller, Fig. 4, and gaining access to shared bus by turning bus connector BC8 off and bus connector BC6 on.

Fig. 6 provides details of the main processor board 13 (13'). Once more, for clarity, address and control signals are deleted, as is some connecting circuitry. The µp2 9 under control of the program stored in ROM2 21 and SPRAM2 28 is responsible for collecting recorded data in 6k blocks from the line interface boards 12 and supplying playback data in 6k blocks to the line interface boards. The µp2 9 transfers data between the line interface boards and an array of eighty 6kx8 RAMs 8 using its DMA2 as shown in Fig. 6. RAM 8, the main memory MM, is organized exactly as the intermediate memory described in Fig. 4, except that it has eight rather than four rows of 64kx1 devices. The µp2 9 also has the task of transferring data -- as usual in 6k blocks -- between the

main memory and the discs 11. Despite its size, the main memory can hold only 2.5 x 80 = 192 seconds or 3.2 minutes of conversation. Thus the task of storage must be passed on to the discs promptly to keep the main memory from filling up.

Using the shared bus approach described in connection with Fig. 5 above, and shown diagramically in Fig. 6 as data transfer means 25, data is either placed into or extracted from one of two 6kx8 RAMs 22 through µp2's data bus DB'. This data is removed and supplied to the disc system or is supplied from the disc system by a further microprocessor µp3 operating under control of the program stored in a ROM3 and SPRAM3. The special requirements of transferring data between one of the RAMs 22 and the discs is met by the use of disc control circuit 32 and high speed disc data bus 34 rather than the DMA devices associated with previously described data transfers. Circuit 32 represents a special version of circuits used to effect such transfers, however, in that it handles data in 6k blocks rather than the much shorter 256 byte blocks conventionally employed.

Finally, µp2 9 is concerned with communication for control and other purposes with the outside world, represented as "other interfaces" 33. For this purpose, ports are provided allowing access to µp2's data bus DB' using the same method as described above with respect to transfer of data between main memory and the intermediate memories of the line processor boards. In particular, external signals are required to inform the system as to which message is to be retrieved, what channel it is to be delivered through, which channel is calling for service to record a message, and how the message is to be identified for later recall. Further it is desireable to back up the contents of the discs on a cassette or other large scale memory device.

A flow chart, simplified for clarity, of the program used to operate the µp1 on each of the line interface boards is presented in Fig. 7. The µp1 maintains a file of eight records in its scratchpad memory, each of which relates to the status of one of the eight input/output channels and -- in the event the channel is not idle -- the status of the message which the channel is either recording or playing back.

Initially, the μpl reads the status of the first channel at A. If the channel is idle, μpl proceeds -- after completing the data transfer task indicated in block B--later to examine the second channel. The μpl continues to examine the status of each channel in order. In the event that all channels are idle, μpl simply cycles through a status check of the channels over and over. The program returns to look at the first channel following a check of the eighth channel; and the idle cycle is shown by the arrow C.

When μpl reads the status of a channel which is not idle, it determines from the status data whether the channel is to be used to record or play back. This information is initially provided by outside control through μp2 and remains fixed until the channel is returned to idle status, again by outside control relayed by μp2.

Once the channel is designated either as a record of playback channel D, upl sets the switch S (and other circuitry) shown in Fig. 3 for the appropriate service. The switch S is shown set at E upon the first cycle through this portion of the program and remains with this setting until the channels' status is changed.

The next time µpl cycles to the line under consideration, if the line is in the record mode, µpl will examine the status of the FIFO memories at F, and determine if the last FIFO is filled, G. If it is not filled, µpl returns to examine the status of the next line; if the last FIFO is filled, µpl transfers the data in as many filled FIFO memories as exist at the time to the first available empty 6k block of IM. The transfer H is effected using either an internal register of µpl or direct memory access, whichever is faster, as described above. This process continues in subsequent cycles until µpl notes that the 6k IM block assigned to the channel is filled (at I). At that time, the filed 6k IM block is scheduled for transfer to the main procesor board and a new empty 6k IM block is assigned to the channel for further data transfers J.

Before examining the next channel for status A, µpl will always check to see if there are any 6k IM blocks of data waiting for transfer at B. If there are (and if the data link to µp2 is available), the 6k IM block at the head of the queue will be placed into the link to await pickup by µp2.

When µpl cycles to examine the status of a channel which is designated as a playback channel, µpl

will examine the status of the FIFOs at K, once the switch S has been properly set at E. If the first FIFO is not empty, μpl returns to examine the status to the next line. If the first FIFO, however, is empty, as at L, μpl transfers data to fill as many empty FIFO memories as exist at that time from the 6k block of IM with the message data being played back (M). The transfer is effected using either an internal register of μpl or direct memory access, whichever is faster, as before described. The process continues in subsequent cycles until μpl notes that the 6k IM block assigned to the channel is empty, as at N. At this time, the next 6k IM block is assigned for further data transfer to the channel and a request for transfer of a new data block is put out to μp2. The μpl keeps at least two blocks of 6k data available in IM for transfer to each channel in playback mode.

Unlike the scheduled transfer of data at B for channels in the record mode, μpl is interrupted by μp2 whenever a block of playback data is placed in the data transfer link by μp2. The μpl services the interrupt by transferring the data out of the link into the first available block of IM. Changes in the

status records initiated by outside commands (begin recording, begin playback, and so forth) are relayed by μp2 by means of interrupts to μp1.

In Fig. 8, a similar flow chart is presented for the program which controls the main processor μp2 on the main processor boards 13. The μp2 maintains 32 records in the illustrative preferred embodiment illustrated and described herein, one for each input/output channel on the (up to) four line interface boards. The program starts by examining the status of the first channel on the first line interface board at A'.If it is idle (B'), the μp2 goes on to examine the second channel on the first line interface board, and so forth. When the system is configured with less than four line interface boards, the status indication for an unpopulated position informs μp2 of its absence and the program immediately proceeds to examine the status of the next channel. After examining the 32nd channel (the eighth channel of the fourth line interface board), the program returns to examine the status of the first channel of the first line interface board. If all channels are either idle or unpopulated, the program continues to cycle as shown by arrow C'.

Commands generated outside the system identify channels which are assigned to record a message and assign the message an identifying number; other commands identify a channel to be used to play back a message and identify the message to be played back by the number assigned to it when it was recorded. The commands are placed in the data link connecting µp2's data bus to a port on the main processor board. Such commands are accompanied by an interrupt to µp2 which then services the command by reading the contents into the appropriate record of the status table.

When the program first examines the status of a channel which has received a command, it first determines whether a record or playback status exists at D'.

If "record" is indicated, the program then determines whether it has recognized this status before or not, at E'. If not, the program puts the new status out to the appropriate µpl by placing the data into the data link for the correct line interface board and issuing an interrupt to the appropriate µpl as at F'. If the data link is not available at this point, the program skips to examine the next channel;

and when it returns, it will again attempt to inform the appropriate µpl. If the program recognizes that it has successfully informed µpl on a previous cycle, it determines whether a 6k block of data from the channel exists yet in the main memory, as at G'. If it does, it is queued for transfer to the discs at H' and a new block is assigned to the channel for subsequent recording. The program then returns to check the status of the next channel.

If "playback" is indicated, on the other hand, the program then determines whether it has recognized this status before or not at I'. If not, the program puts the new status out to the appropriate µpl by placing the data into the data link for the correct line interface board and issuing an interrupt to the appropriate µpl at step J'. If the data link is not available at this point, the program skips to examine the next channel; and when it returns, it will again attempt to inform the appropriate µpl. If the program recognizes that it has successfully informed µpl on a previous cycle, it determines whether the channel is calling for more data. If so, and if a 6k MM block is full (steps K' and L'), the 6k block is sent out to

the appropriate line interface board at M'. The program then asks if at least two blocks of 6k MM data remain awaiting transfer at N'. If not, the program requests a 6k block data from the discs at O'. The program then returns to examine the next channel's status.

Further modifications will occur to those skilled in this art, including other types of memory, data transfer and control techniques useable to practice the method of the invention, and also other types of communication serviceable by the same; and such are considered to fall within the spirit and scope of the invention as defined in the appended claims.

CLAIMS

1. A method of message communication digitized storage (in recording mode) and forwarding (in playback mode) along a plurality of input/output channels with simultaneous access to multiple users with the aid of a plurality of line interface systems having series/parallel first-in-first-out memories (FIFO) and intermediate memory RAMS under first microprocessor control and with data transfer connection to main memory RAMS the contents of which are data transferable under second microprocessor control with memory discs, said method comprising, reading the message recording or playback status of each of the plurality of channels in order; if in a recording mode, determining the state of filling of the FIFO memories and, under the control of the first microprocessor, transferring the data in as many filled FIFO memories as exist at such time to the first first available empty data block of the intermediate memory RAMS, and when such block is filled, transferring the data

playback mode, determining the state of filling of the successive FIFO memories and transferring channel data blocks to fill empty FIFO memories with message data being played back; examining the recording or playback status of each input/output channel of the plurality of line interface systems under control of the second microprocessor; receiving commands identifying which channels are assigned for recording and which for playback; if a recording mode is assigned, and the said examining of said status has recognized such recording mode, determining whether data blocks from such channel still exist in the main memory and transferring such to the memory discs; if the recording modes is not recognized during such examining, communicating such status to the first microprocessor of the appropriate channel line interface system; if a playback mode is assigned, and the said examining of such status has recognized such playback mode, determining whether the channel is calling for more data, and if so, transferring data

blocks of such channel from full main memory RAMs to intermediate memory RAMs and successive FIFO memories of the appropriate channel line interface system for playback; and, if the playback mode is not recognized during such examining, communicating an interrupt to the first microprocessor of the appropriate line interface system.

2. A method as claimed in claim 1 and in which the messages are voice messages received from users and digitized for recording, and converted back from digital to voice analog form for playback transmission to the users.

3. A method as claimed in claim 1 and in which said transferring of data is effected using either the internal register of the microprocessor or direct memory access, whichever is faster.

4. A method as claimed in claim 1 and in which the further step is performed in playback mode, upon the determination that the channel data block in the said intermediate memory is empty, of assigning the next data block for

further data transfer to the channel and requesting the main memory microprocessor to transfer a new data block, such that at least two data blocks are available in said intermediate memory for transfer to each channel in playback mode.

5. A method as claimed in claim 1 and in which in the event that a channel is idle during the said reading of the message recording or playback status, the first microprocessor proceeds to read the next channel status, cycling through the channels in the event they are idle until reading the status of a channel which is not idle.

6. A method as claimed in claim 1 and in which, during the recording mode, the first microprocessor is interrupted by the main memory second microprocessor whenever a block of playback data is placed in the data transfer link by said second microprocesor; the first microprocessor servicing the interrupt by transferring data out of the link into the first available block of said intermediate memory.

7. A method as claimed in claim 2 and in which the said data block is selected as substantially a 6k block of data.

8. A method as claimed in claim 2 and in which at least eight input/output channels are provided per line interface system with four FIFO memories, at least four such systems for each main memory, and with the intermediate memory comprising forty 6k x 8 RAMs and the main memory eighty 6K x 8 RAMs.

9. A method of message communication digitized storage (in recording mode) and forwarding (in playback mode) along a plurality of input/output channels with simultaneous access to multiple users with the aid of a plurality of microprocessor-controlled line-interface systems having series/parallel first-in-first-out memories (FIFO) and intermediate memory RAMs with data transfer connection to main memory RAMs, the contents of which are data transferable under further microprocessor control, with memory discs; said method comprising, in recording mode, applying digitized bit data representing the received message to the top register of the first FIFO memory and, when filled, rippling

the contents immediately through the FIFO memories to the bottom register of the last FIFO memory, with continuation of this rippling so as to fill the lower FIFO memories but without filling the first FIFO memory; assigning a particular message to be recorded to an intermediate memory RAM to be filled; at irregular intervals, when at least the lowermost FIFO memory is filled, transferring the contents of the bottom FIFO memory to the top RAM of the said intermediate memory RAM, and continuing such transfer such that the oldest data is at the bottom of the FIFO memories and the oldest data is at the top of the intermediate memory RAMs; queuing the filled intermediate memory RAMs and, at irregular intervals, transferring data in a queued intermediate memory RAM to main memory RAM and thence to a recording disc; in playback mode, finding the segment of the disc containng the first part of the message to be played back and, at irregular intervals, applying the data thereof to an available main memory RAM and then to an available intermediate memory RAM in the appropriate line interface system for playback of this

message; transferring the message data in the top register of such appropriate RAM to the top register of the first of the FIFO memories; rippling the same down through the FIFO memories to the lowermost unfilled register of the last FIFO memory; removing the contents of such lowermost register and transmitting the same in form suitable for playback transmission to the user, such that the series of FIFO memories is never completely empty in order to avoid delays in the played back messages that are long enough to be perceived by the user.

10. In a method of message communication digitized storage (in recording mode) and forwarding (in playback mode) along a plurality of input/output channels with simultaneous access to multiple users with the aid of a plurality of microprocessor-controlled line-interface systems having series/parallel first-in-first-out memories (FIFO) and intermediate memory RAMs wih data transfer connection to main memory RAMs, the contents of

which are data transferable, under further microprocessor control, with memory discs; the method of message recording that comprises, applying digitized bit data representing a received message to the top register of the first FIFO memory and, when filled, rippling the contents immediately through the FIFO memories to the bottom register of the last FIFO memory, with continuation of this rippling so as to fill the lower FIFO memories but without filling the first FIFO memory; assigning a particular message to be recorded to an intermediate memory RAM to be filled; at irregular intervals, when at least the lowermost FIFO memory is filled, transferring the contents of the bottom FIFO memory to the top RAM of the said intermediate memory RAM, and continuing such transfer such that the oldest data is at the bottom of the FIFO memories and the oldest data is at the top of the intermediate memory RAMs; queuing the filled intermediate memory RAMs and, at irregular intervals, transferring data in a queued intermediate memory RAM to main memory RAM and thence to a recording disc.

11. In a method of message communication digitized storage (in recording mode) and forwarding (in playback mode) along a plurality of input/output channels with simultaneous access to multiple users with the aid of a plurality of microprocessor-controlled line-interface systems having series/parallel first-in-first-out memories (FIFO) and intermediate memory RAMs with data transfer connection to main memory RAMs, the contents of which are data transferable, under further microprocessor control, with memory discs; the method of playback that comprises, finding the segment of the disc containing the first part of a recorded data bit message to be played back and, at irregular intervals, applying the data thereof to an available main memory RAM and thence to an available intermediate memory RAM in the appropriate line interface system for playback of this message; transferring the message data in the top register of such appropriate RAM to the top register of the first of the FIFO memor-

ies; rippling the same down through the FIFO memories to the lowermost unfilled register of the last FIFO memory; removing the contents of such lowermost register and transmitting the same in form suitable for playback transmission to the user, such that the series of FIFO memories is never completely empty in order to avoid delays in the played back messages that are long enough to be perceived by the user.

12. A method as claimed in claim 9 and in which said communication messages are voice messages and the same are received and digitized for the said applying step in said recording mode; and, following the said step of removing the contents of said lowermost FIFO register memory during playback mode, converting the digital message data back to analog voice communication form for the said transmission to the user.

13. Apparatus for message storage and forwarding with simultaneous access to multiple sources having, in combination, means operable

in recording mode for inputting messages in bit streams of digital data; means for applying such bit streams to a plurality of bit series/parallel first-in-first-out (FIFO) memories to assemble therein multiple-bit words from the data stream; means for shifting contents from the top register of the first of such memories, when filled, down the memories to the last available unfilled register of the memories; first microprocessor means for monitoring the state of filling of the FIFO memories; means operable when at least the last of the FIFO memories is filled for transferring the contents thereof to intermediate memories of the random access type (RAM) with the oldest data in the lowermost FIFO memories transferred to the topmost of the RAM, and such that the first FIFO never completely fills; means operable after such transfer for rippling the data in the partially filled FIFO memories, not transferred, to the lowermost FIFO memory for continuation of the data filling process; means

for storing the transferred bits of data in the RAM intermediate memories; main memory processor means having a second microprocessor means for examining the RAM contents, queuing the data-filled RAMs, and transferring the date to main memory RAMs; means controlled by the second microprocessor means for transferring the data of filled RAMs of the main memory, at irregular intervals, under the control of disc control means comprising a third microprocessor means, to record the same as successive segments on disc means; in playback mode, means controlled by the third microprocessor means for locating the first part of the message stored in the disc means to be played back; means operable at irregular inervals under the control of the said second microprocessor means for assigning an available main memory RAM to receive the message data located in the disc means and for recognizing the intermediate memory RAM containing the communication channel to which the playback is directed; means for transferring the message data from the

assigned main memory RAM to the recognized intermediate memory RAM, with the first message data to be played back being transferred to the top register of the appropriate intermediate memory RAM; means for transferring the message data in the said top register of the intermediate memory RAM to the top register of the first of said FIFO memories and rippling the same down through the FIFO memories to the lowermost available register of the lowermost FIFO memory; means for removing the contents of the said lowermost register and transmitting the same to the user, but such that the FIFO memories are never completely empty, so as to avoid delays in the played back messages that are long enough to be perceived by the user; the last-named means operating so that succeeeding segments of data are transferred to an intermediate memory RAM by the time the transfer of the previous segment is complete, insuring continuity of message communication.

14. Apparatus for voice communication message storage and forwarding with simultaneous access to multiple sources having, in combination, means operable in recording mode for receiving and converting voice messages into bit streams of digital data; means for applying such bit streams to a plurality of bit series/parallel first-in-first-out (FIFO) memories to assemble therein multiple-bit words from the data stream; means for shifting contents from the top register of the first of such memories, when filled, down the memories to the last available unfilled register of the memories; first microprocessor means for monitoring the state of filling of the FIFO memories; means operable when at least the last of the FIFO memories is filled for transferring the contents thereof to intermediate memories of the random access type (RAM) with the oldest data in the lowermost FIFO memories transferred to the topmost of the RAM, and such that the first FIFO never completely fills; means operable after

such transfer for rippling the date in the partially filled FIFO memories, not transferred, to the lowermost FIFO memory for continuation of the data filling process; means for storing the transferred bits of data, representing the received voice communication messages, in the RAM intermediate memories; main memory processor means having a second microprocessor means for examining the RAM contents, queuing the data-filled RAMs, and transferring the data to main memory RAMs; means controlled by the second microprocessor means for transferring the data of filled RAMs of the main memory at irregular intervals under the control of disc control means comprising a third microproocesor means, to record the same as successive segments on disc means; in playback mode, means controlled by the third microprocessor means for locating the first part of the communication message stored in the disc means to be played back; means operable at irregular intervals under the control of the said second microprocessor means for assigning an available main memory RAM to receive the message data

located in the disc means and for recognizing the intermediate memory RAM containing the communication channel to which the playback is directed; means for transferring the message data from the assigned main memory RAM to the recognized intermediate memory RAM, with the first message data to be played back being transferred to the top register of the appropriate intermediate memory RAM; means for transferring the message data in the said top register of the intermediate memory RAM to the top register of the first of said FIFO memories and rippling the same down through the FIFO memories to the lowermost available register of the lowermost FIFO memory; means for removing the contents of the said lowermost register and applying the same to digital-to-analog converter means to transmit the voice analog communication message to the user, but such that the FIFO memories are never completely empty, so as to avoid delays in the played back communication messages that are long enough to be perceived by the

user; the last-named means operating so that succeeding segments of data are transferred to an intermediate memory RAM by the time the transfer of the previous segment is complete, insuring continuity of voice communication.

15. Apparatus for voice communication message storage for later forwarding with simultaneous access to multiple sources having, in combination, means operable in recording mode for receiving and converting voice messages into bit streams of digital data; means for applying such bit streams into a plurality of bit series/parallel first-in-first-out (FIFO) memories to assemble therein multiple-bit words from the data stream; means for shifting contents from the top register of the first of such memories, when filled, down the memories to the last available unfilled register of the memories; first microprocessor means for monitoring the state of filling of the FIFO memories; means operable when at least the last of the FIFO memories is filled for transferring the contents thereof to intermediate memories of the random access

type (RAM) with the oldest data in the lowermost FIFO memories transferred to the topmost of the RAM, and such that the first FIFO never completely fills; means operable after such transfer for rippling the data in the partially filled FIFO memories, not transferred, to the lowermost FIFO memory for continuation of the data filling process; means for storing the transferred bits of data representing the received voice communication messages in the RAM intermediate memories; and main memory processor means having a second microprocessor means for examining the RAM contents, queuing the data-filled RAMs, and transferring the data to main memory RAMs; means controlled by the second microprocessor means for transferring the data of filled RAMs of the main memory at irregular intervals under the control of disc control means comprising a third microprocessor means, to record the same as successive segments on disc means.

16. In apparatus for voice communiation storage and forwarding with simultaneous access to multiple sources wherein a plurality of intermediate memories of the random access type (RAM) have received bit data streams

representing digitized voice messages from first-in-first-out (FIFO) intermediate memories and have transferred the same through main memory RAMs to record the same as successive segments on disc means; apparatus for playback mode operation having, in combination, means controlled by a third microprocessor means for locating the first part of a communication message stored in the disc means to be played back; means operable at irregular intervals under the control of a second microprocesor means for assigning an available main memory RAM to receive the message data located in the disc means and for recognizing the intermediate memory RAM containing the communication channel to which the playback is directed; means for transferring the message data from the assigned main memory RAM to the recognized intermediate memory RAM, with the first message data to be played back being transferred to the top register of the appropriate intermediate memory RAM; means for transferring the message data in the said top register of the intermediate memory RAM to the top register

of the first of said FIFO memories and rippling the same down through the FIFO memories to the lowermost available register of the lowermost FIFO memory; means for removing the contents of the said lowermost register and applying the same to digital-to-analog converter means to transmit the voice analog communication message to the user, but such that the FIFO memories are never completely empty, so as to avoid delays in the played back communication messages that are long enough to be perceived by the user; the last-named means operating so that succeeding segments of data are transferred to an intermediate memory RAM by the time the transfer of the previous segment is complete, insuring continuity of voice communication.

17. Apparatus as claimed in claim 14 and in which each microprocessor means is provided with read-only memory and scratchpad means, that of the first microprocessor means enabling the said monitoring of the state of filling of the FIFO memories, that of the second microprocessor means controlling the said

examining of RAM contents, and that of the third microprocessor means for said locating and assigning steps.

18. Apparatus as claimed in claim 14 and in which the said converting means is of the continuously variable slope delta modulation type in which the continuously variable voice signal is converted into a bit stream, reconstitutable in playback mode by said digital-to-analog converter means as said voice signal.

19. Apparatus as claimed in claim 14 and in which the digital data is provided in substantially 6k blocks of data, and at least eight input/output channels are provided for each of four sets of four FIFO and forty 6k x 8 intermediate RAM memory systems for each main memory system, and the main memory is provided with eighty 6k x 8 RAMs.

FIG. 1.

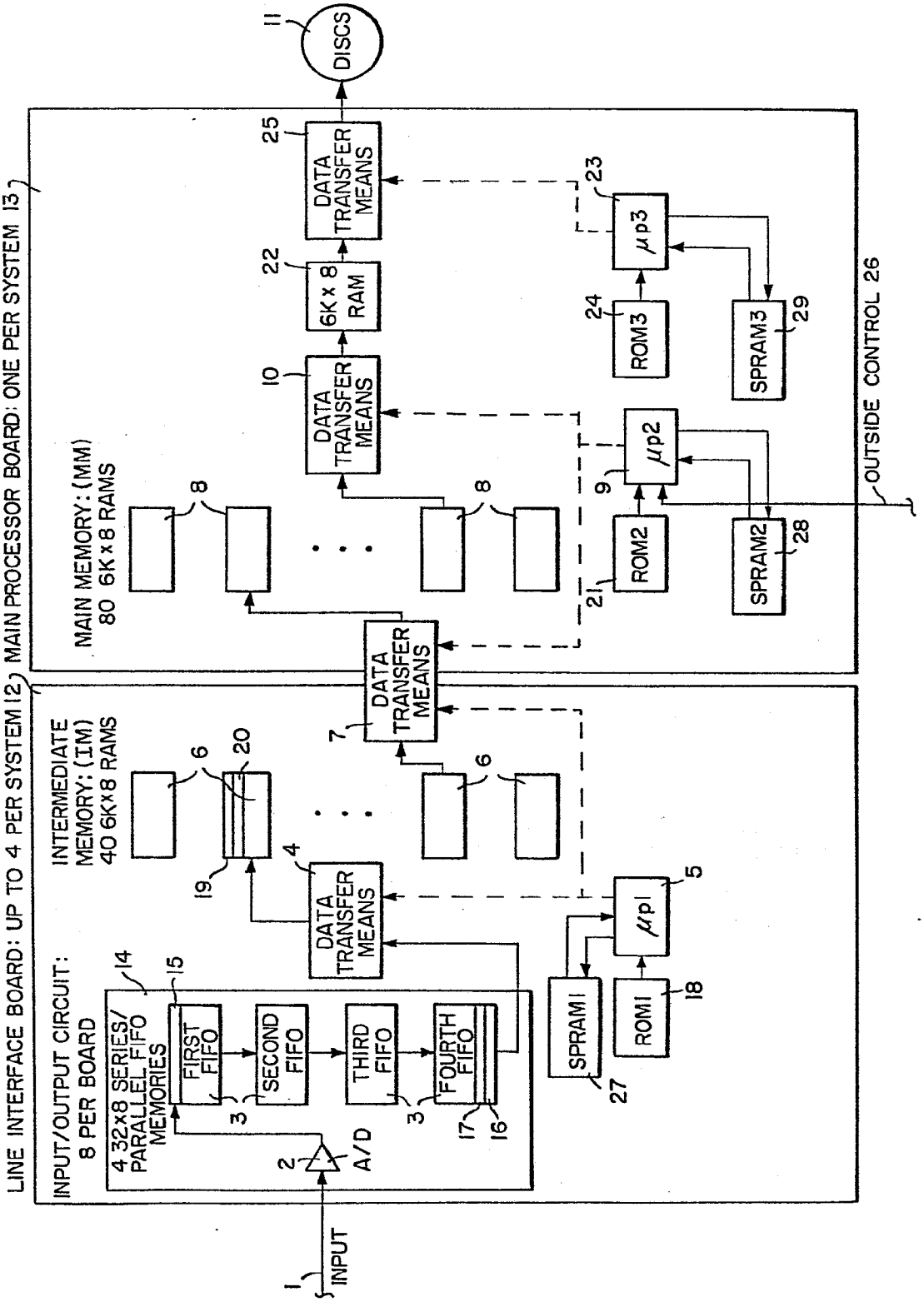

LINE INTERFACE BOARD: UP TO 4 PER SYSTEM 12
INPUT/OUTPUT CIRCUIT: 8 PER BOARD
4 32x8 SERIES/ PARALLEL FIFO MEMORIES
FIRST FIFO
SECOND FIFO
THIRD FIFO
FOURTH FIFO
A/D
INPUT
INTERMEDIATE MEMORY: (IM) 40 6K x 8 RAMS
DATA TRANSFER MEANS
μp1
SPRAM1
ROM1
MAIN PROCESSOR BOARD: ONE PER SYSTEM 13
MAIN MEMORY: (MM) 80 6K x 8 RAMS
6K x 8 RAM
μp2
ROM2
SPRAM2
μp3
ROM3
SPRAM3
OUTSIDE CONTROL 26
DISCS

FIG. 2.

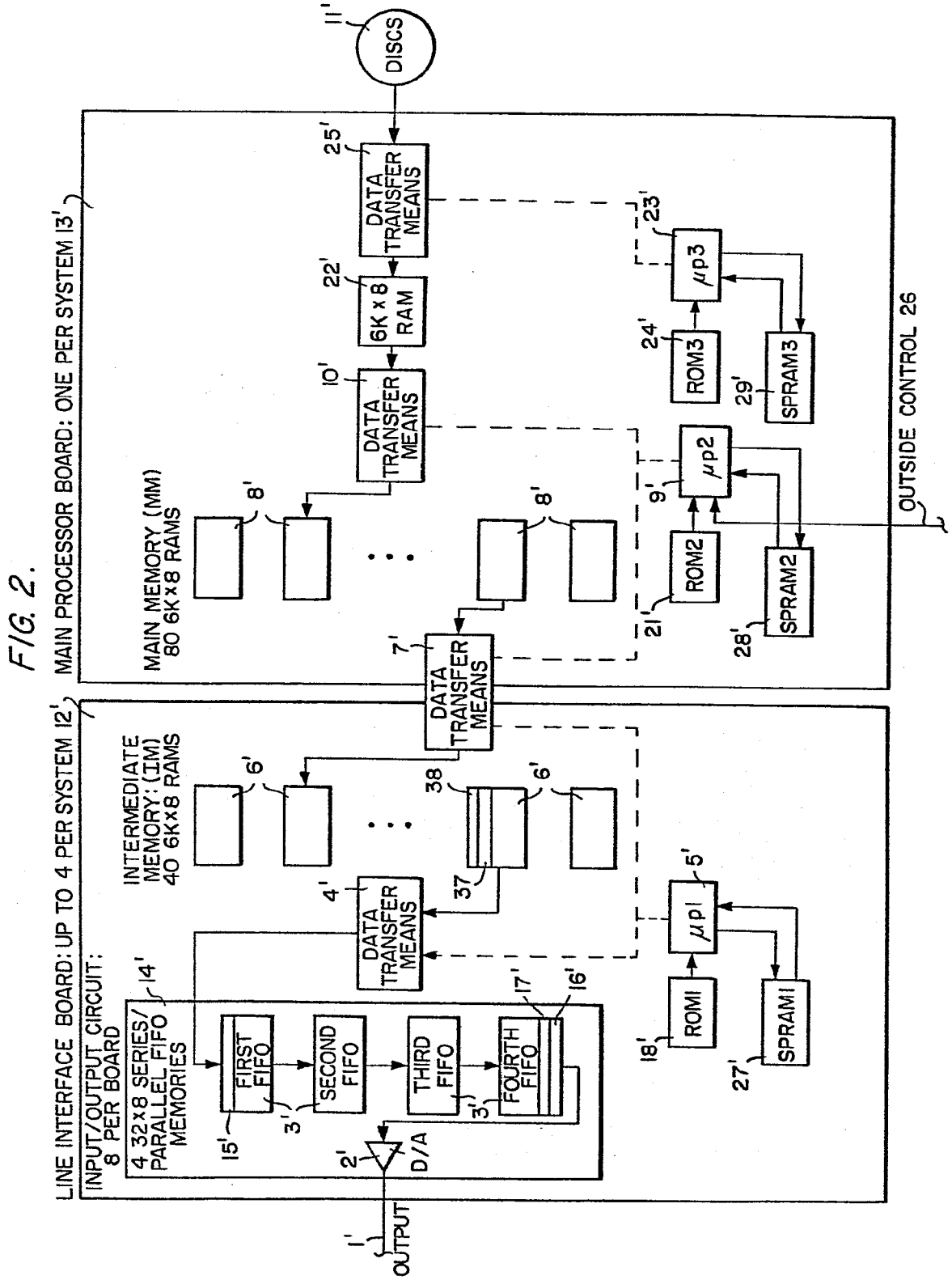
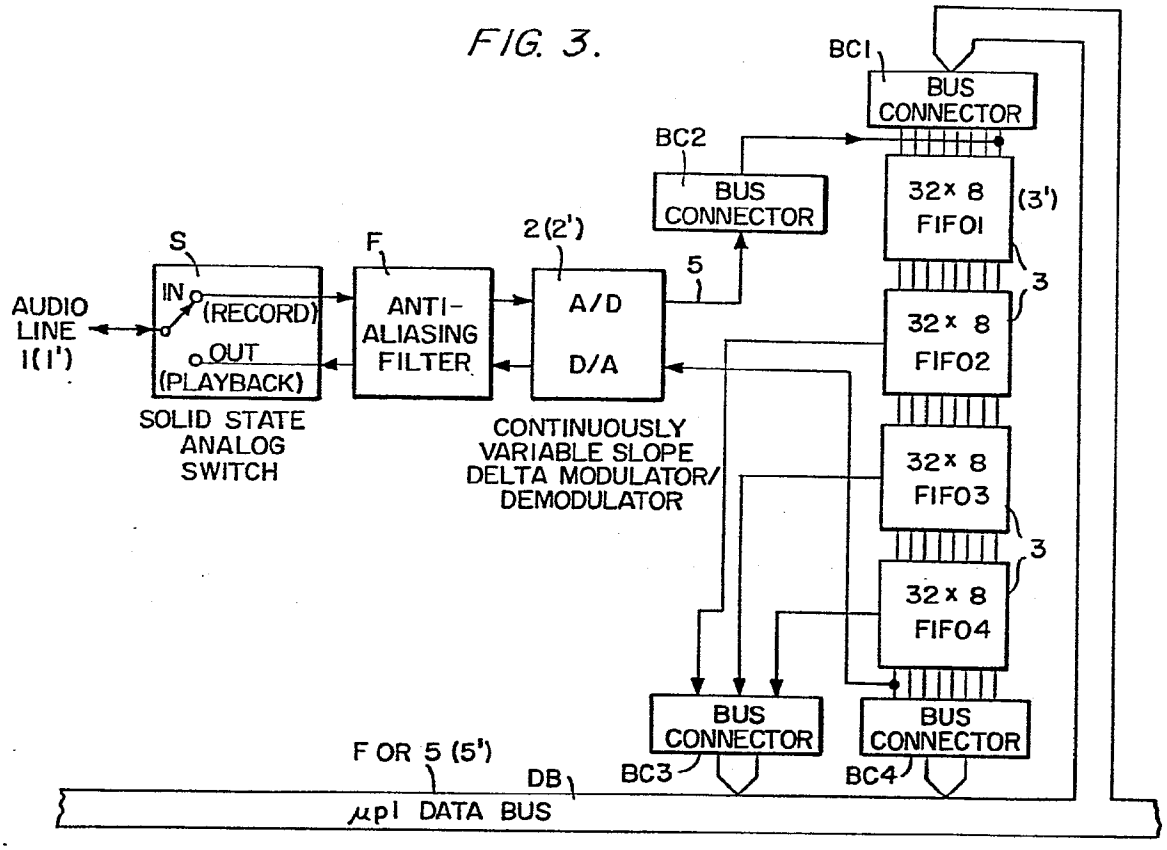

DISCS
11'
MAIN PROCESSOR BOARD: ONE PER SYSTEM 13'
DATA TRANSFER MEANS
25'
6K x 8 RAM
22'
DATA TRANSFER MEANS
10'
MAIN MEMORY (MM) 80 6K x 8 RAMS
8'
DATA TRANSFER MEANS
7'
OUTSIDE CONTROL 26
μp3
23'
ROM3
24'
SPRAM3
29'
μp2
9'
ROM2
21'
SPRAM2
28'
LINE INTERFACE BOARD: UP TO 4 PER SYSTEM 12'
INTERMEDIATE MEMORY: (IM) 40 6K x 8 RAMS
6'
38
37
DATA TRANSFER MEANS
4'
INPUT/OUTPUT CIRCUIT: 8 PER BOARD
4 32x8 SERIES/ PARALLEL FIFO MEMORIES
14'
FIRST FIFO
15'
SECOND FIFO
3'
THIRD FIFO
3'
FOURTH FIFO
17'
16'
D/A
2'
OUTPUT
1'
μp1
5'
ROMI
18'
SPRAMI
27'

FIG. 3.


BC1
BUS CONNECTOR
BC2
BUS CONNECTOR
32x 8 FIFO1
(3')
3
32x 8 FIFO2
32x 8 FIFO3
3
32x 8 FIFO4
S
IN
(RECORD)
OUT
(PLAYBACK)
AUDIO LINE 1(1')
SOLID STATE ANALOG SWITCH
F
ANTI-ALIASING FILTER
2(2')
A/D
D/A
5
CONTINUOUSLY VARIABLE SLOPE DELTA MODULATOR/ DEMODULATOR
BUS CONNECTOR
BUS CONNECTOR
BC3
BC4
F OR 5 (5')
DB
µpI DATA BUS

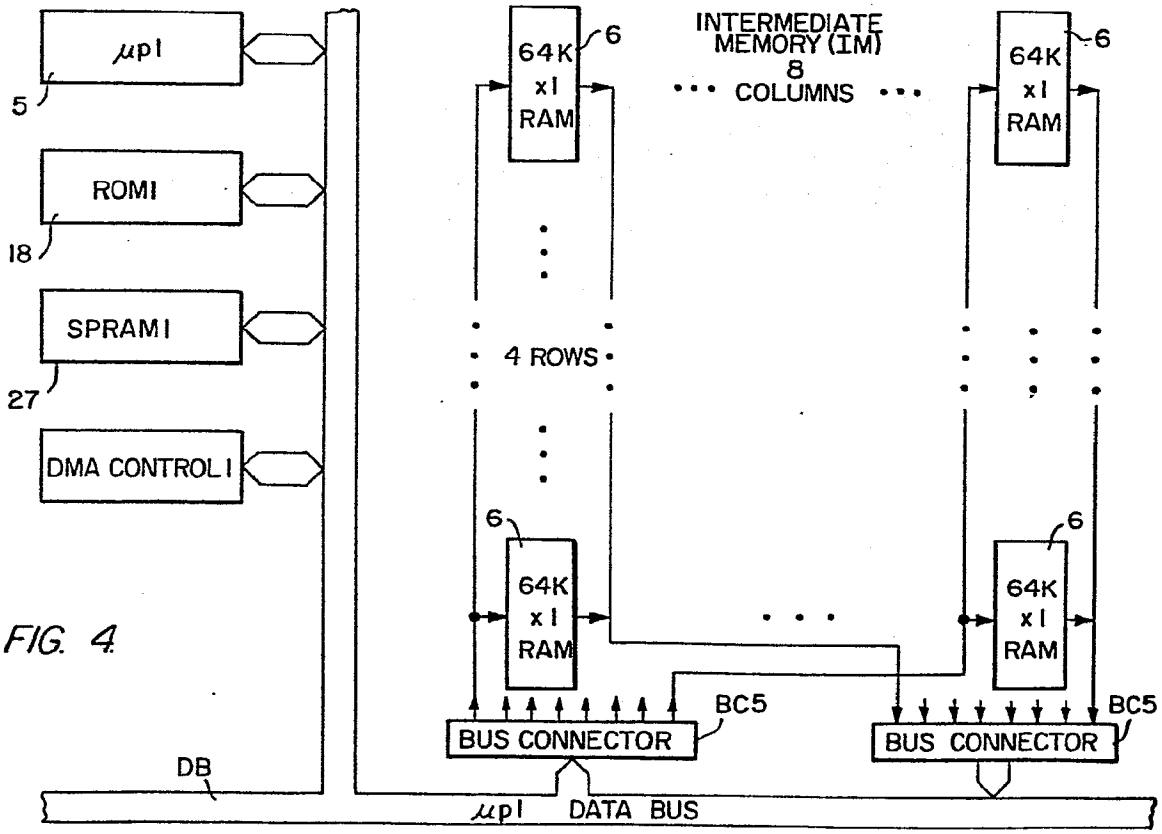

µpI
5
ROMI
18
SPRAMI
27
DMA CONTROLI
64K x1 RAM
6
INTERMEDIATE MEMORY (IM)
8
COLUMNS
4 ROWS
BC5
BUS CONNECTOR
DB
µpI DATA BUS

FIG. 4

FIG. 5.

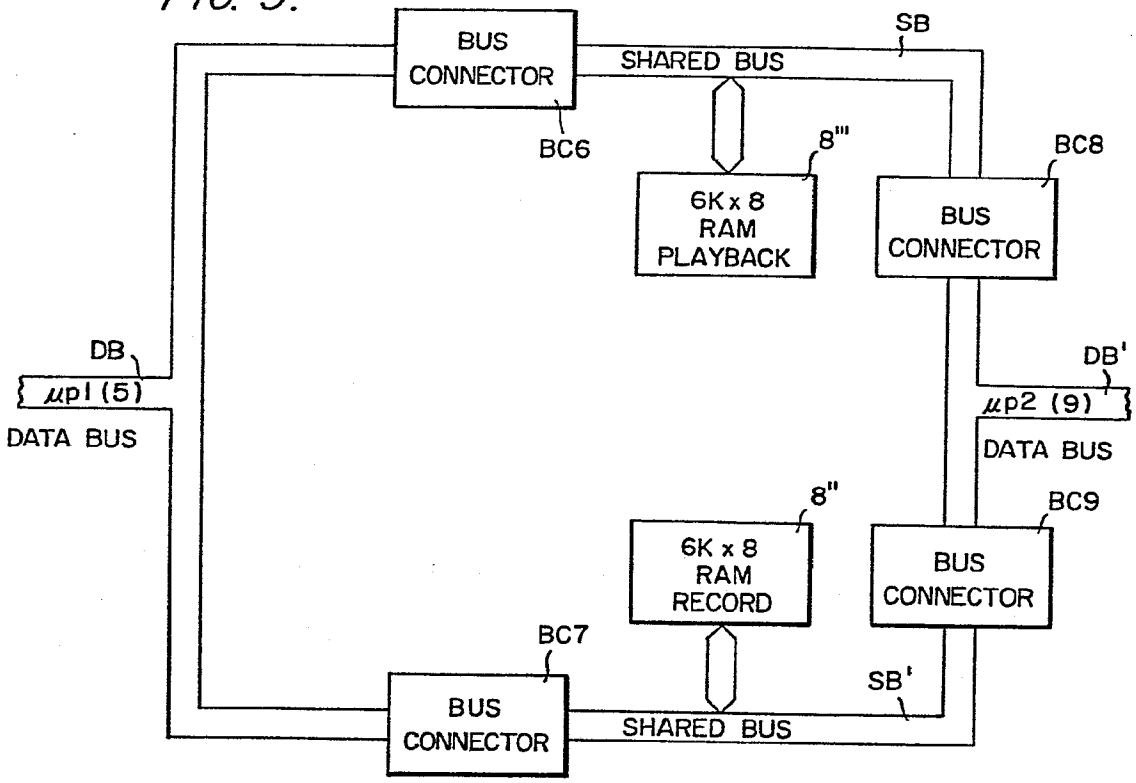

BUS CONNECTOR
SHARED BUS
SB
BC6
8'''
6K x 8 RAM PLAYBACK
BC8
BUS CONNECTOR
DB
μp1 (5)
DATA BUS
DB'
μp2 (9)
DATA BUS
8''
6K x 8 RAM RECORD
BC9
BUS CONNECTOR
BC7
BUS CONNECTOR
SB'
SHARED BUS

FIG. 6.

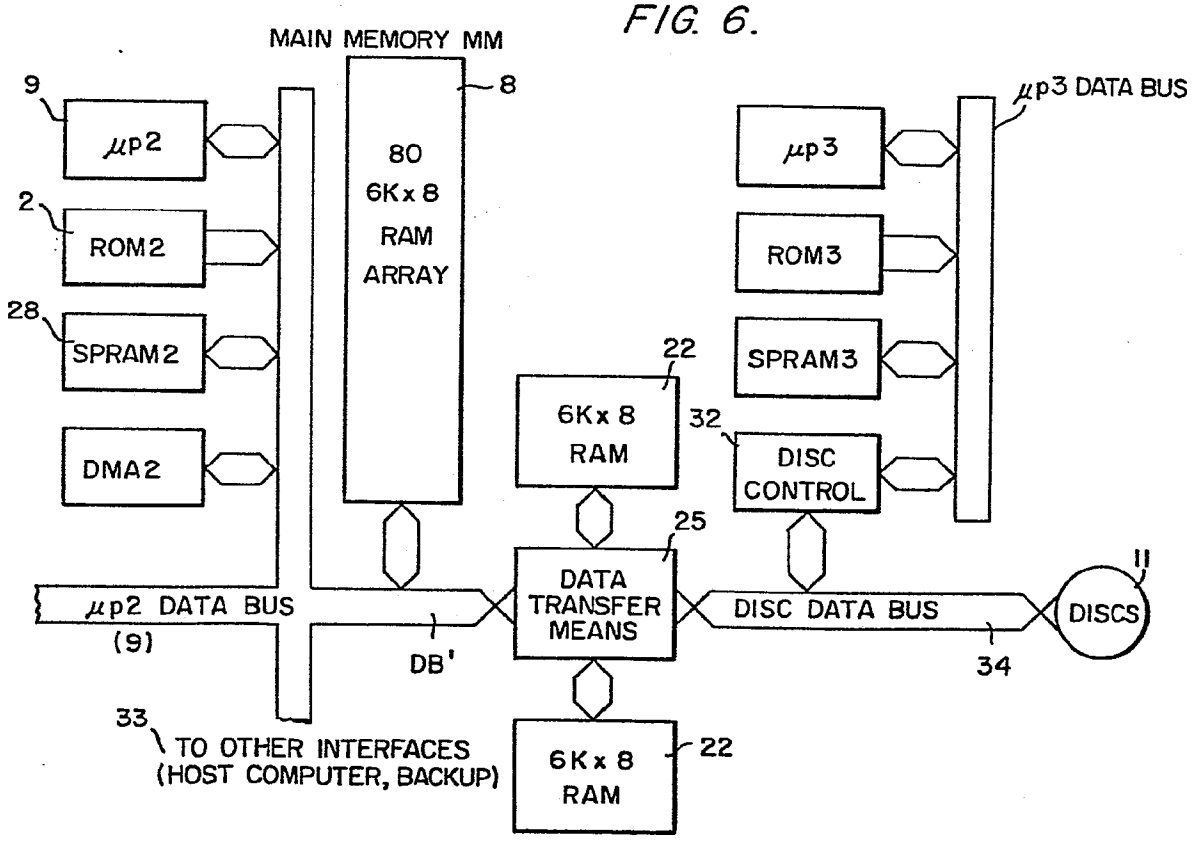

MAIN MEMORY MM
9
μp2
2
ROM2
28
SPRAM2
DMA2
80 6K x 8 RAM ARRAY
8
μp3 DATA BUS
μp3
ROM3
SPRAM3
22
6K x 8 RAM
32
DISC CONTROL
25
DATA TRANSFER MEANS
11
DISCS
μp2 DATA BUS
(9)
DB'
DISC DATA BUS
34
33
TO OTHER INTERFACES (HOST COMPUTER, BACKUP)
6K x 8 RAM
22

FIG. 7.

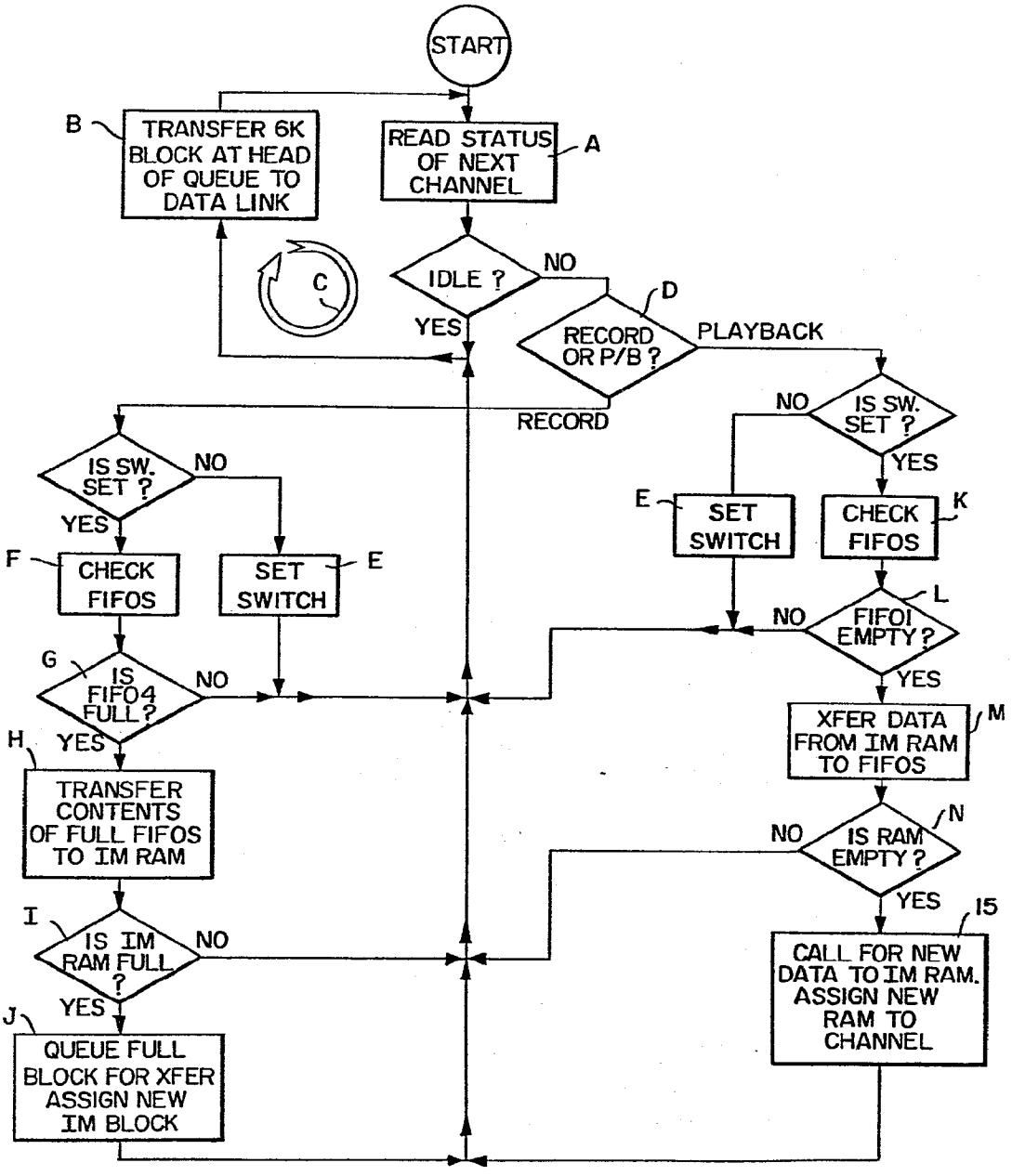
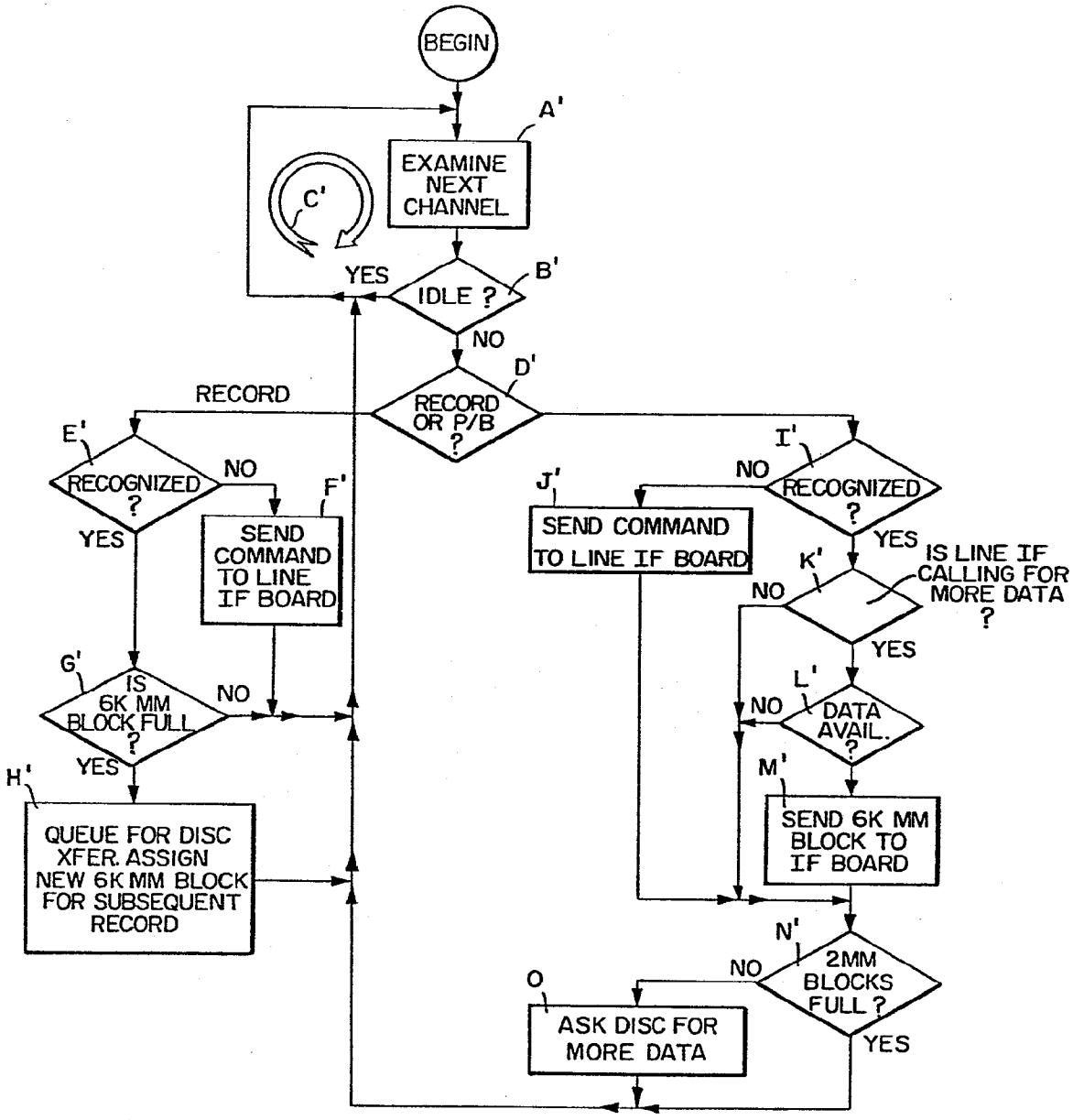

START
READ STATUS OF NEXT CHANNEL
A
IDLE ?
NO
YES
C
B
TRANSFER 6K BLOCK AT HEAD OF QUEUE TO DATA LINK
D
RECORD OR P/B ?
PLAYBACK
RECORD
IS SW. SET ?
NO
YES
F
CHECK FIFOS
SET SWITCH
E
G
IS FIFO4 FULL?
NO
YES
H
TRANSFER CONTENTS OF FULL FIFOS TO IM RAM
I
IS IM RAM FULL ?
NO
YES
J
QUEUE FULL BLOCK FOR XFER ASSIGN NEW IM BLOCK
IS SW. SET ?
NO
YES
E
SET SWITCH
CHECK FIFOS
K
L
FIFO1 EMPTY?
NO
YES
M
XFER DATA FROM IM RAM TO FIFOS
N
IS RAM EMPTY ?
NO
YES
I5
CALL FOR NEW DATA TO IM RAM. ASSIGN NEW RAM TO CHANNEL

FIG. 8.


BEGIN
A'
EXAMINE NEXT CHANNEL
C'
YES
IDLE ?
B'
NO
D'
RECORD OR P/B ?
RECORD
E'
RECOGNIZED ?
NO
F'
SEND COMMAND TO LINE IF BOARD
YES
G'
IS 6K MM BLOCK FULL ?
NO
YES
H'
QUEUE FOR DISC XFER. ASSIGN NEW 6K MM BLOCK FOR SUBSEQUENT RECORD
I'
NO
RECOGNIZED ?
YES
J'
SEND COMMAND TO LINE IF BOARD
K'
IS LINE IF CALLING FOR MORE DATA ?
NO
YES
L'
NO
DATA AVAIL. ?
M'
SEND 6K MM BLOCK TO IF BOARD
N'
NO
2MM BLOCKS FULL ?
YES
O
ASK DISC FOR MORE DATA